**Escuela Colombiana de Ingeniería Julio Garavito**

**Carrera / Semestre:** Ingeniería de Sistemas / 5to Semestre

**Asignatura:** Modelos y Bases de Datos (MBDA)

**Estudiantes:** Andrés Chavarro, Jesús Pinzón

**Fecha:** 2024/02/22

# Autoestudio 2 – SQL Básico

**Investigación**

1. **NULL**
2. ¿Qué significa?

**NULL** es un tipo de dato que no se encuentra definido o iniciado, como si de un espacio en blanco se tratara, incluso este por sí mismo no es equivalente a otro tipo de dato, siquiera a sí mismo. En otras palabras, es un dato “desconocido” pero no vacío. Además, podemos realizar operaciones con distintos tipos de operadores, permitiendo evitar posibles errores en consultas.

1. ¿Resultado de operarlo con los diferentes tipos de operadores: aritméticos, de comparación y lógicos?

Por el lado de la **aritmética matemática**, se puede tomar el valor **NULL** como la representación de un **cero** como, por ejemplo:

**Suma y Resta:**

**Multiplicación:**

**División:**

Por otro lado, en las **operaciones de** **comparación**, **NULL** suele obtener valores de **UNKNOWN**, a excepción de comparar , por ejemplo:

**Desigualdad:**

**Comparación:**

Por último, en las **operaciones lógicas** un dato **NULL** tiene por definición un **valor desconocido** que da como resultado un valor igualmente desconocido, por ejemplo:

**AND:**

**OR:**

**NOT:**

1. **JUNTA**
2. ¿Cuáles son las diferencias entre junta interna y externa?

Las juntas **(JOINS)** en SQL son operaciones que permiten combinar filas de dos o más tablas basándose en una condición de coincidencia común. Se utilizan para extraer datos que se encuentran distribuidos en diferentes tablas de una manera relacionada. Los tipos principales de juntas incluyen la **INNER JOIN** (junta interna) y **OUTER JOIN** (junta externa), cuyas diferencias se describen en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Característica** | **INNER JOIN** | **OUTER JOIN** |
| **Coincidencias** | Solo devuelve filas cuando hay al menos una coincidencia en ambas tablas. | Devuelve todas las filas de una o ambas tablas, coincidan o no. |
| **Resultados** | Si no hay coincidencia, la fila no se incluye en el resultado. | Incluye filas sin coincidencia, marcando con **NULL** los valores faltantes en las columnas de la tabla sin coincidencias. |
| **Tipos** | Solo hay un tipo. | Incluye **LEFT OUTER JOIN**, **RIGHT OUTER JOIN** y **FULL OUTER JOIN**. |
| **Uso** | Útil cuando el interés está en las filas que tienen correspondencia entre tablas. | Útil para mantener todas las filas de una o ambas tablas, mostrando las relaciones donde existen y marcando con **NULL** donde no. |
| **Filas con NULL** | No incluye filas con **NULL** a menos que la coincidencia incluya valores **NULL** que satisfagan la condición de unión. | Puede incluir filas con **NULL** en las columnas de la tabla sin coincidencias. |

1. ¿Qué opciones se tienen para la junta interna?

Para realizar una junta interna **(INNER JOIN)** en SQL, existen varias opciones o sintaxis que se pueden utilizar, dependiendo de la complejidad de la consulta y las preferencias de desarrollo. Algunas de estas opciones incluyen:

* **Sintaxis Explícita de INNER JOIN:** Es la forma más común y recomendada de realizar una junta interna. Utiliza la cláusula INNER JOIN, seguida por la condición de unión especificada en la cláusula **ON**.
* **Sintaxis Implícita de INNER JOIN:** Antes de la estandarización de la sintaxis de **JOIN**, las juntas internas se realizaban colocando la condición de unión en la cláusula **WHERE**. Aunque todavía es soportada por muchos sistemas de gestión de bases de datos, su uso no es recomendado por cuestiones de claridad y mantenibilidad.
* **Uso de Alias para las Tablas:** Tanto en la sintaxis explícita como en la implícita, es común utilizar alias para las tablas con el din de simplificar las consultas, especialmente cuando se trabaja con nombres de tablas largos o cuando se realizan múltiples juntas.
* **INNER JOIN con Múltiples Tablas:** Es posible unir más de dos tablas en una sola consulta usando múltiples cláusulas INNER JOIN, cada una con su propia condición de unión.
* **INNER JOIN con Condiciones Adicionales:** Además de la condición de unión, se puede agregar condiciones adicionales en la cláusula **ON** o en una cláusula **WHERE** para filtrar aún más los resultados.

1. ¿Qué opciones se tienen para la junta externa?

Para realizar juntas externas **(OUTER JOINS)** en SQL, se tienen principalmente tres opciones, cada una adaptada para incluir filas de una o ambas tablas que no tienen coincidencias directas. Estas opciones son:

### **LEFT OUTER JOIN (Junta Externa Izquierda):** Devuelve todas las filas de la tabla de la izquierda (la primera tabla en la consulta) y las filas coincidentes de la tabla de la derecha. Si no hay coincidencia, los resultados para la tabla de la derecha tendrán **NULL**.

### **RIGHT OUTER JOIN (Junta Externa Derecha):** Devuelve todas las filas de la tabla de la derecha (la segunda tabla en la consulta) y las filas coincidentes de la tabla de la izquierda. Si no hay coincidencia, los resultados para la tabla de la izquierda tendrán **NULL**.

### **FULL OUTER JOIN (Junta Externa Completa):** Combina los resultados de los **LEFT** y **RIGHT OUTER JOINS**. Devuelve todas las filas cuando hay una coincidencia en una de las tablas. Si no hay coincidencia en una de las tablas, el conjunto de resultados incluirá **NULL** en las columnas de la tabla sin coincidencia.

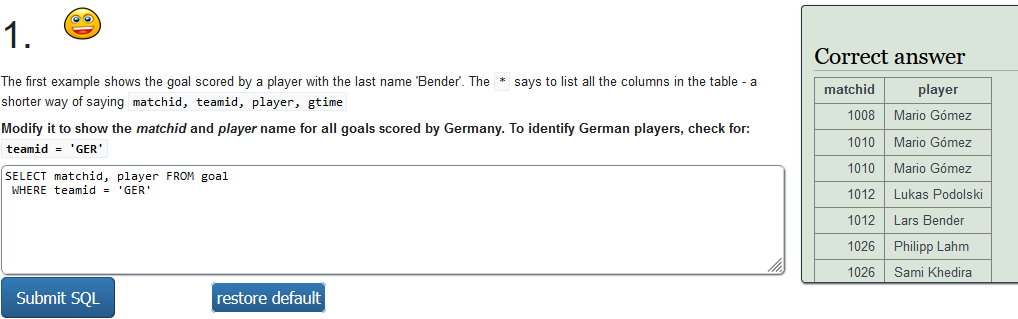
### **Consideraciones Adicionales:**

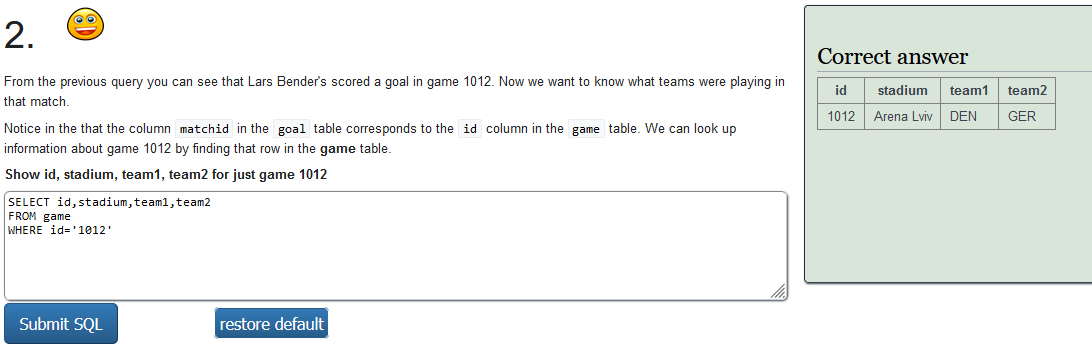
* + **Uso de Alias:** Al igual que con las juntas internas, es común usar alias para las tablas para hacer la consulta más legible, especialmente cuando se trabaja con nombres de tablas largos o se realiza una junta entre múltiples tablas.
  + **Condiciones Adicionales:** Puedes incluir condiciones adicionales en la cláusula **ON** o mediante una cláusula **WHERE** para filtrar aún más los resultados. Esto es particularmente útil para refinar los resultados de una junta externa.
  + **Cruce de Tablas (CROSS JOIN):** Aunque no es una junta externa, el CROSS JOIN merece una mención como otra opción de junta en SQL. Realiza un producto cartesiano entre las filas de las tablas participantes, resultando en todas las posibles combinaciones de filas entre las tablas.

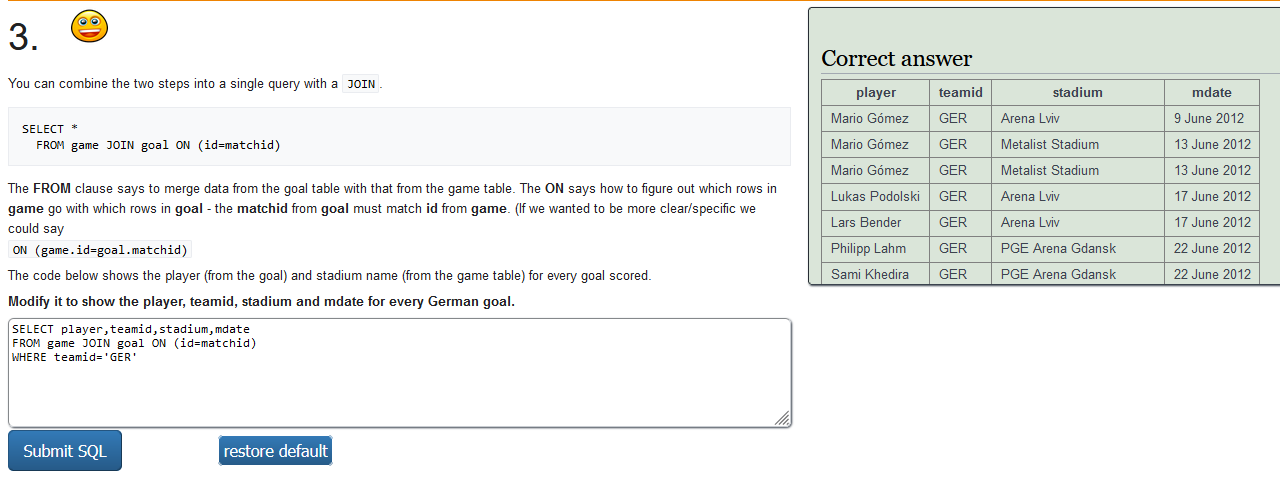
**Práctica**

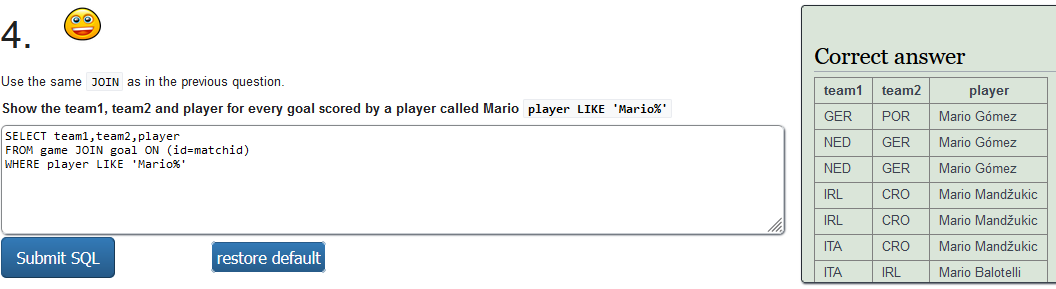
1. Realizar los ejercicios propuestos en los siguientes tutoriales.

* **JOIN:**

****

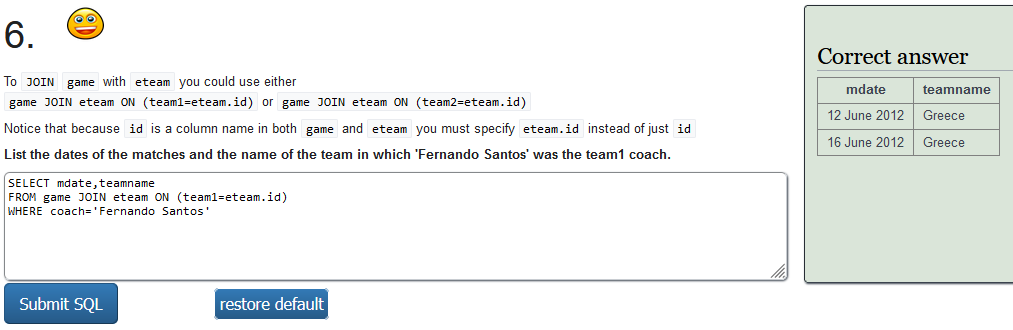
****

****

****

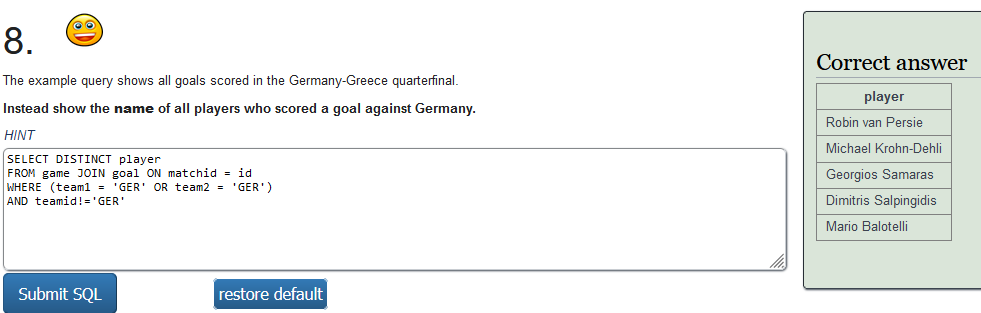
**A screenshot of a computer

Description automatically generated**

****

**A close-up of a computer screen

Description automatically generated**

****

**A white rectangle with black lines

Description automatically generated**

**A white rectangular object with a black border

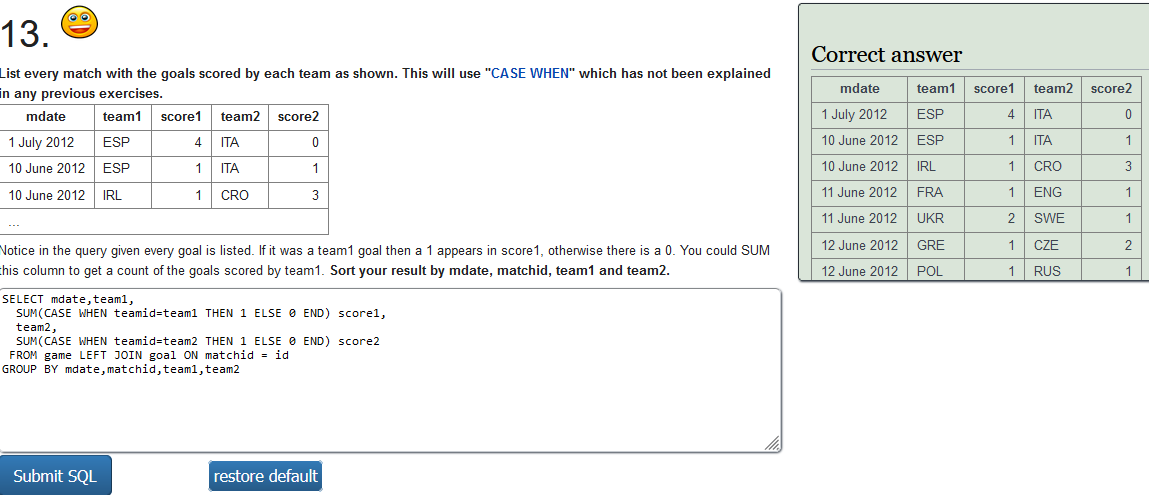
Description automatically generated**

**A close-up of a white rectangular object

Description automatically generated**

**A white rectangular box with black text

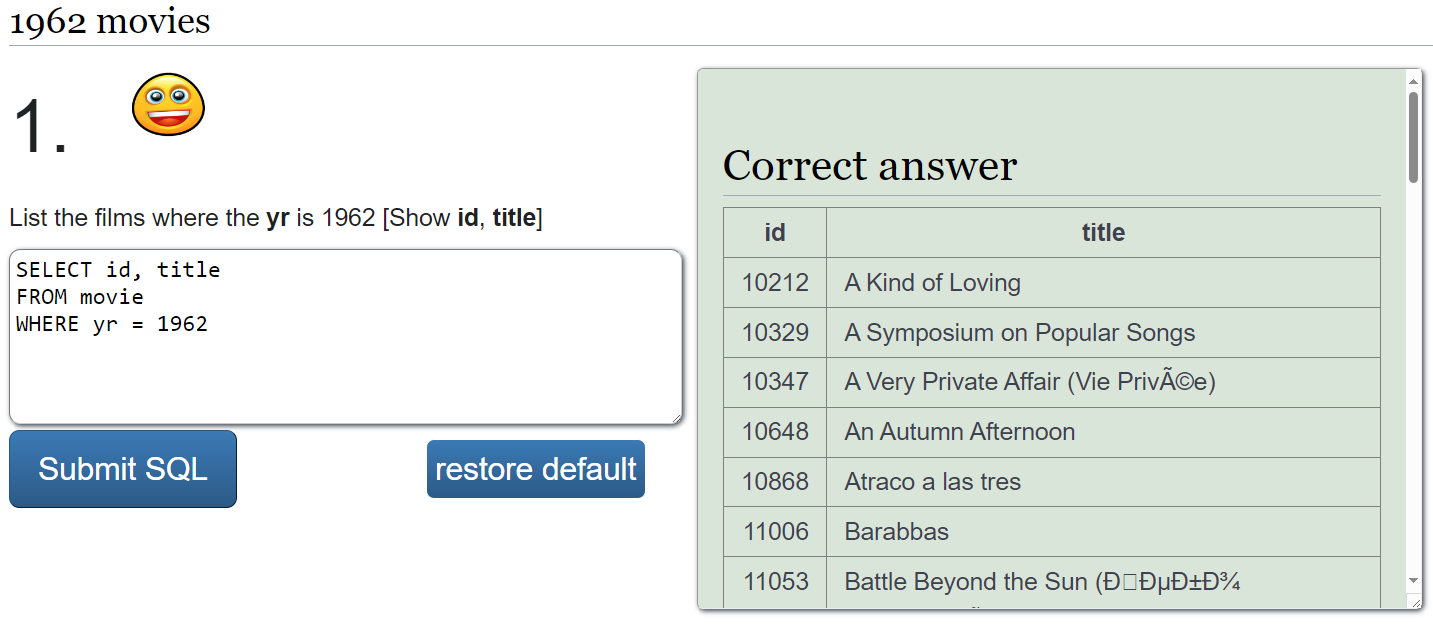
Description automatically generated**

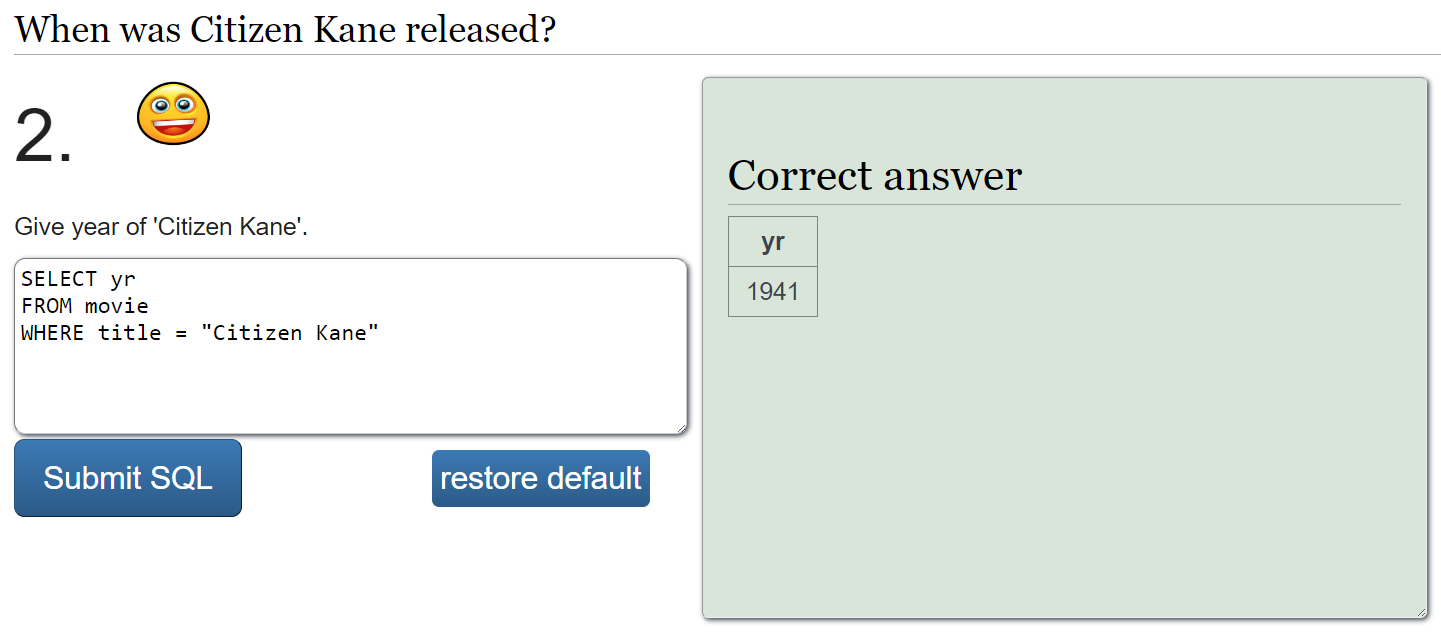
****

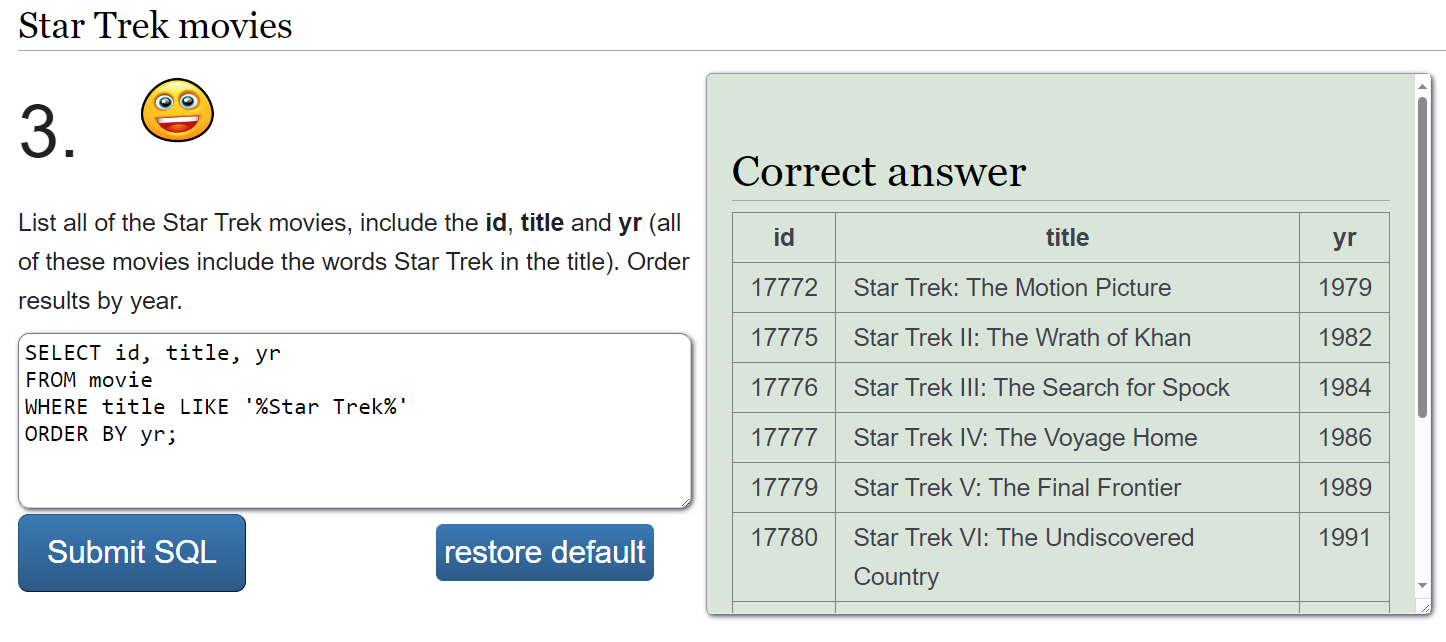
* **Quiz Join:**

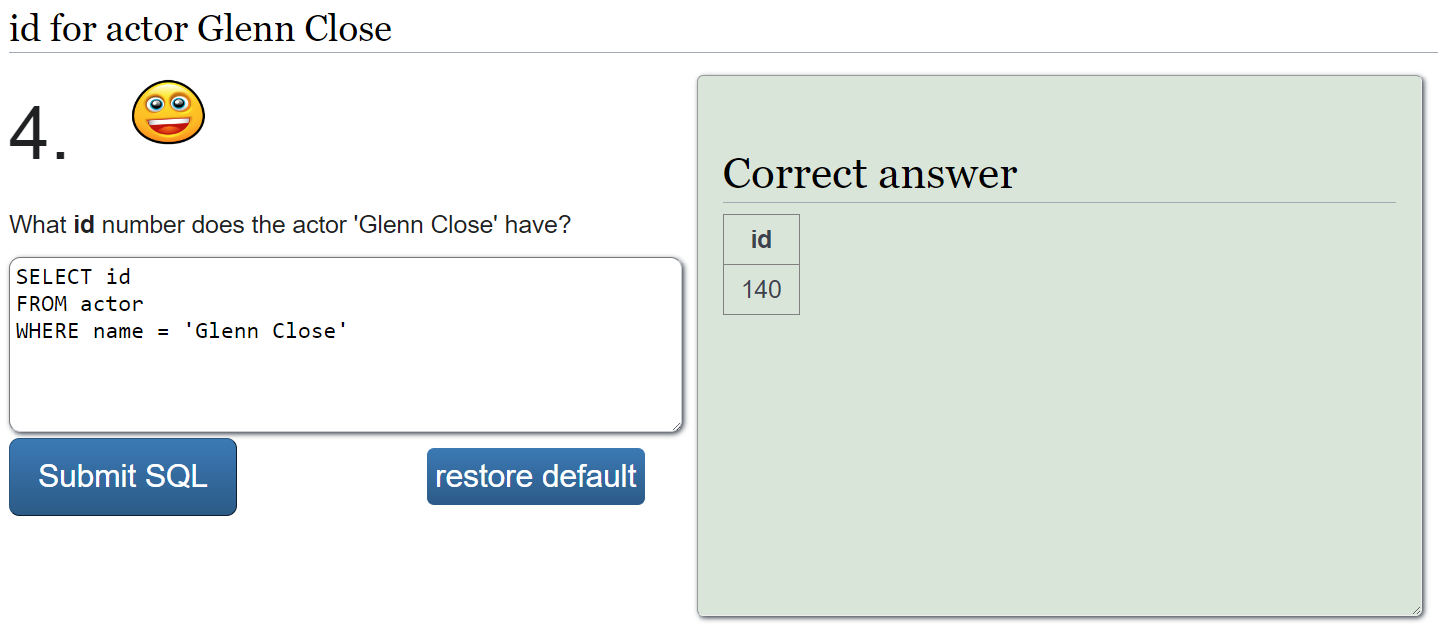
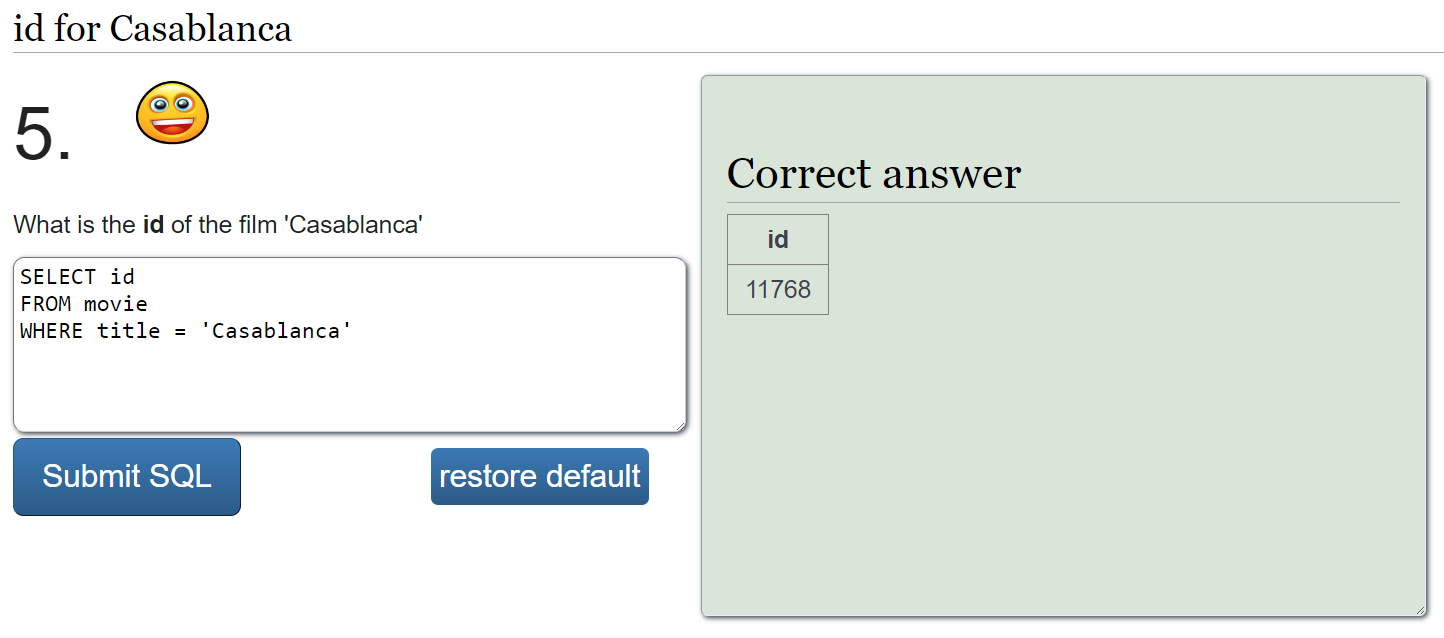
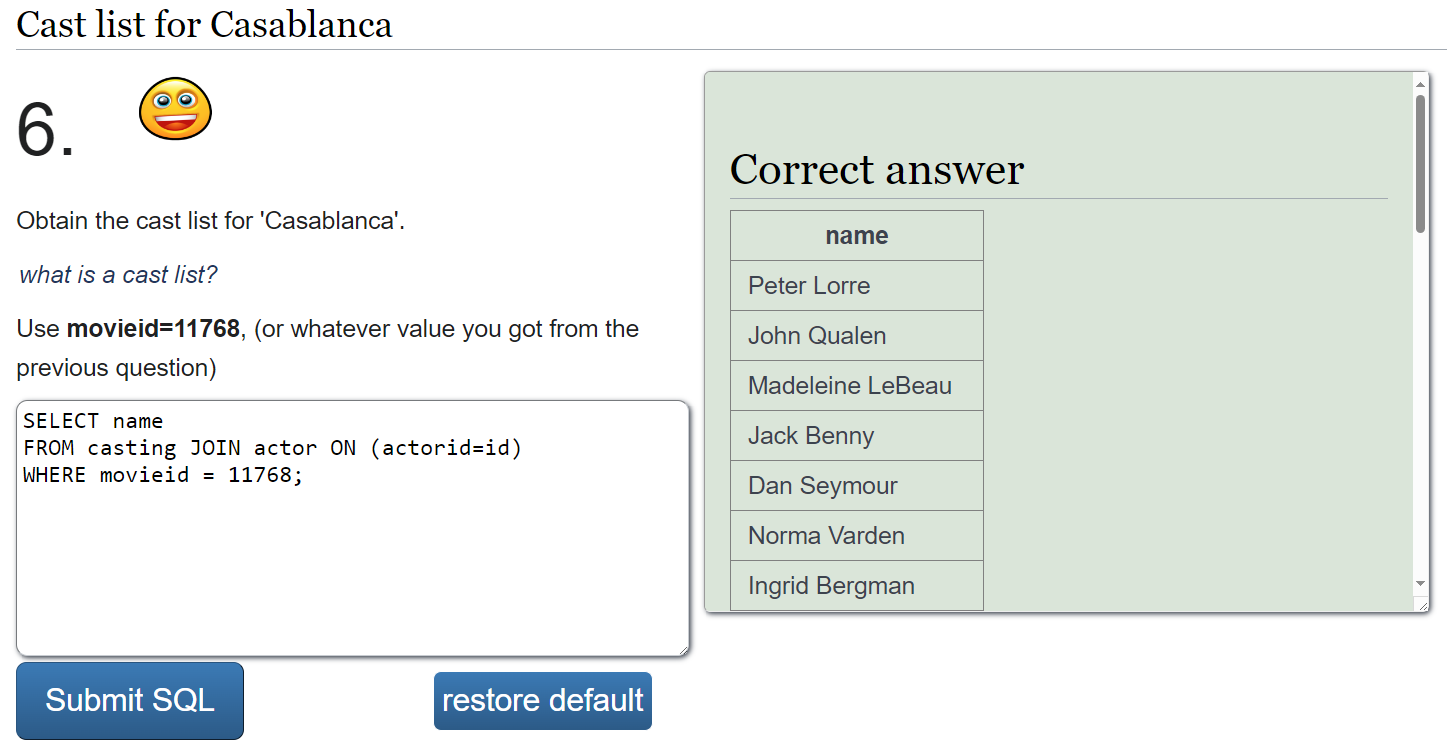
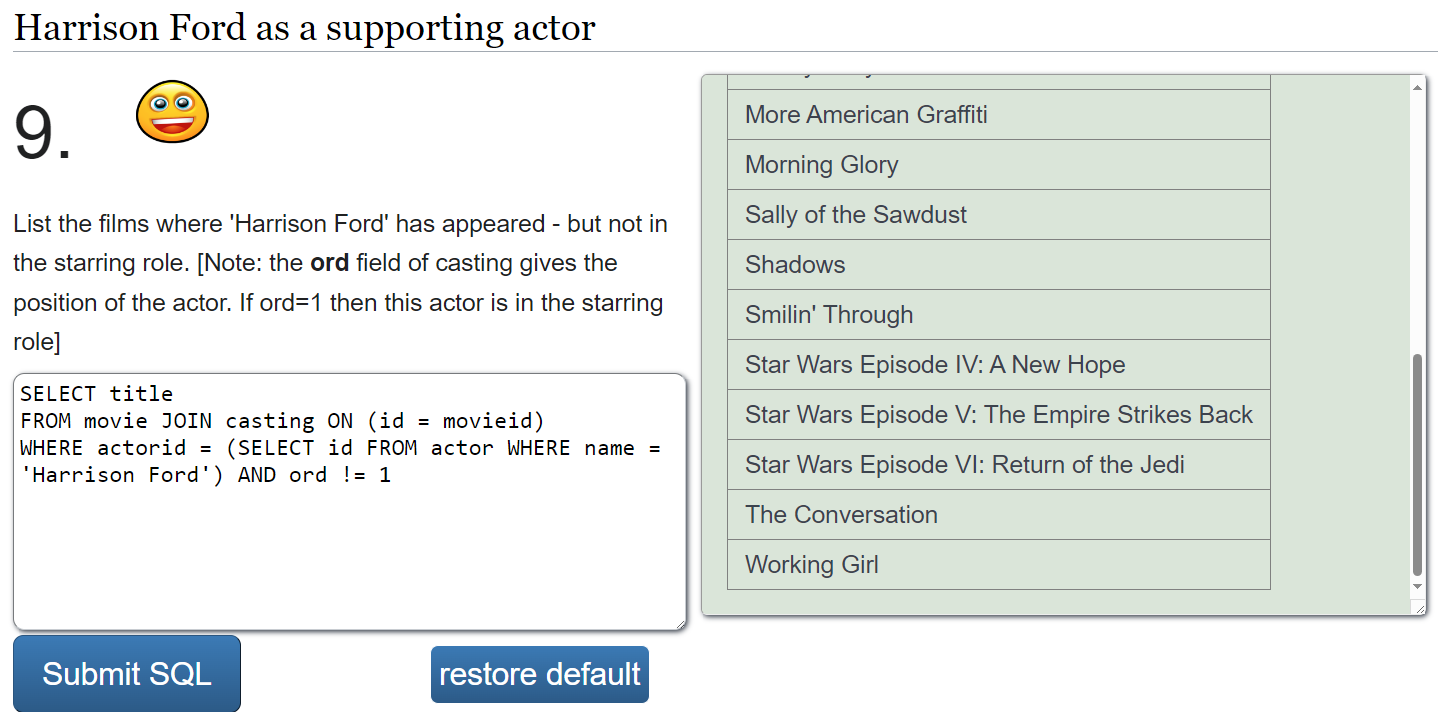
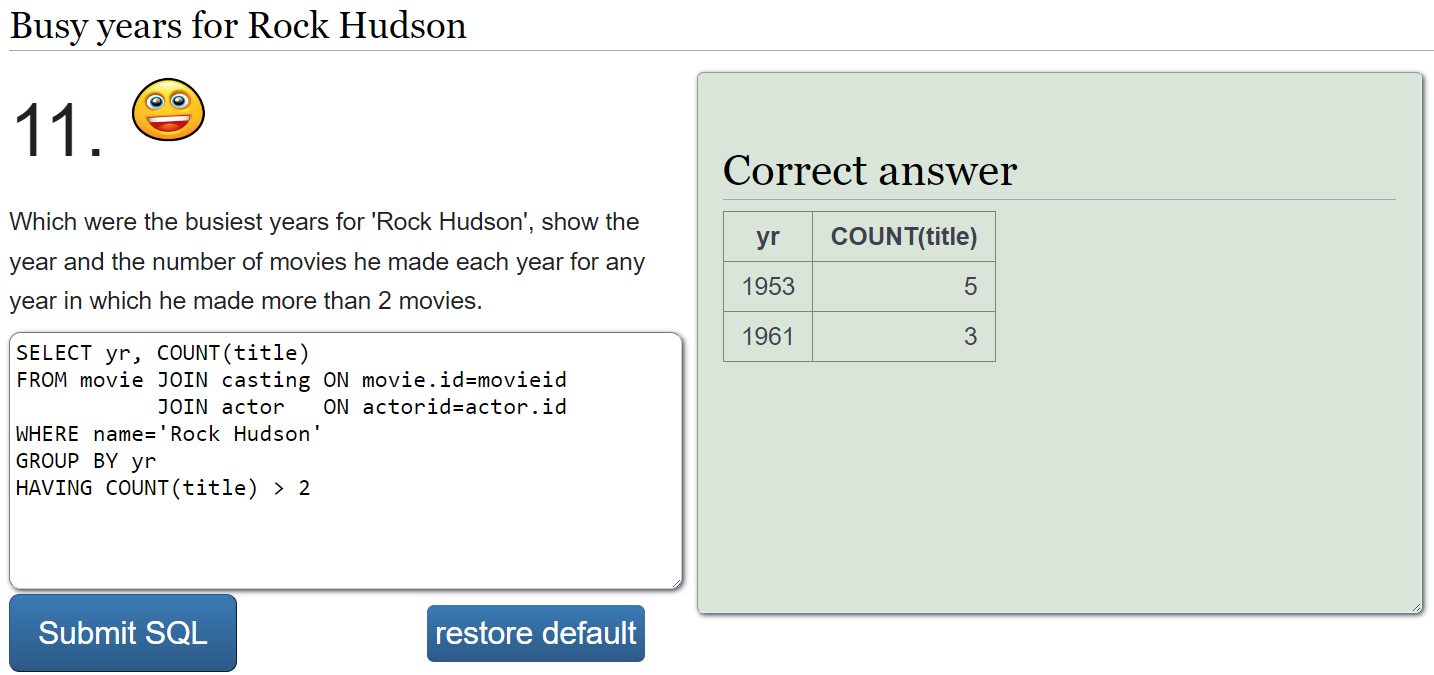


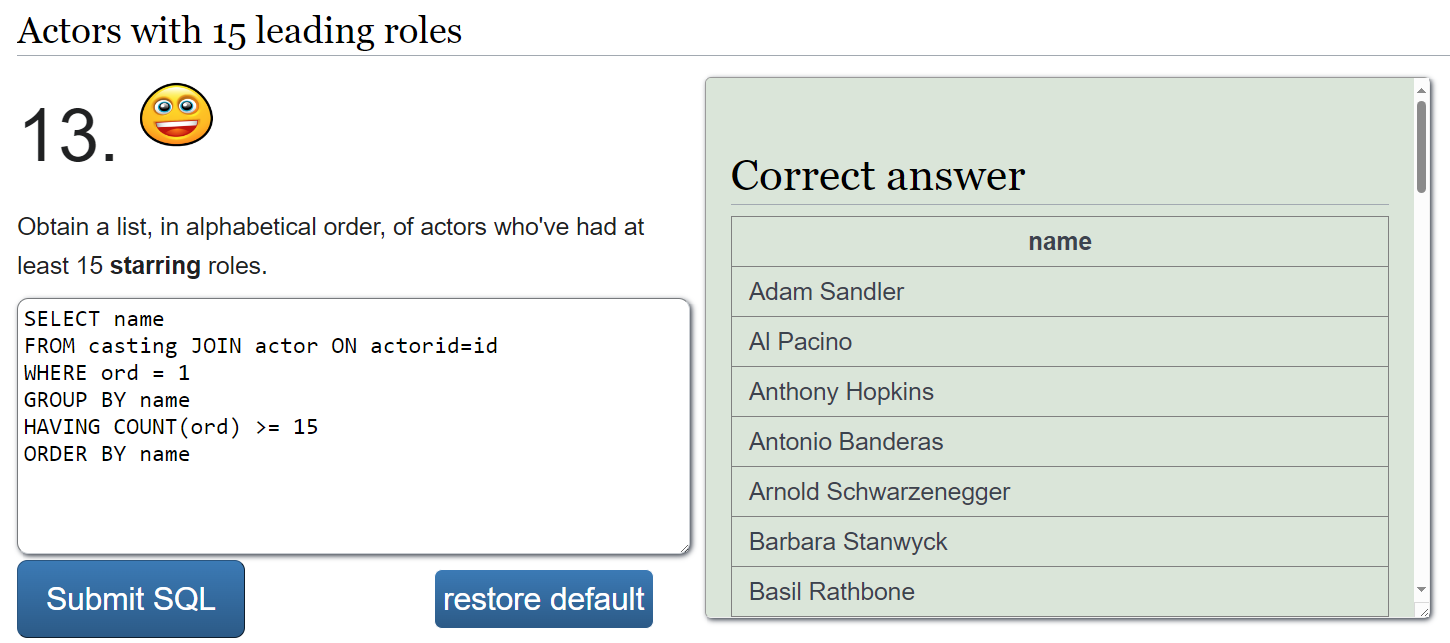
* **More JOIN Operations:**









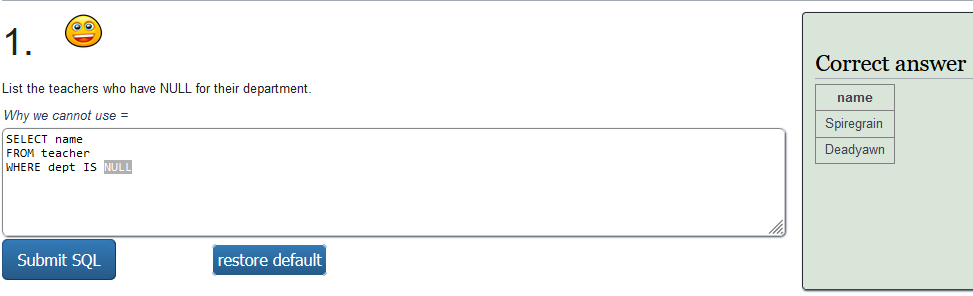


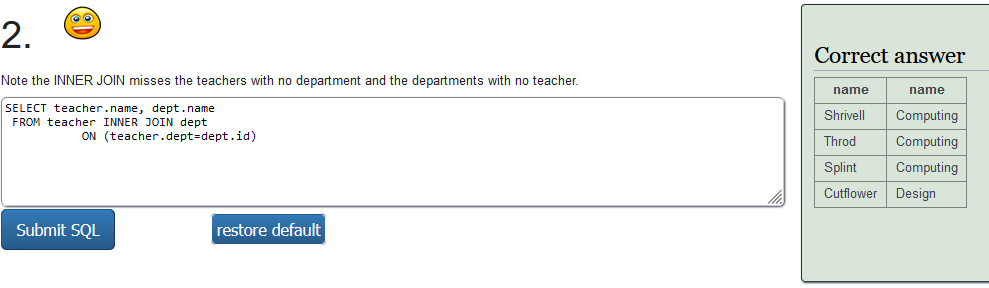
.

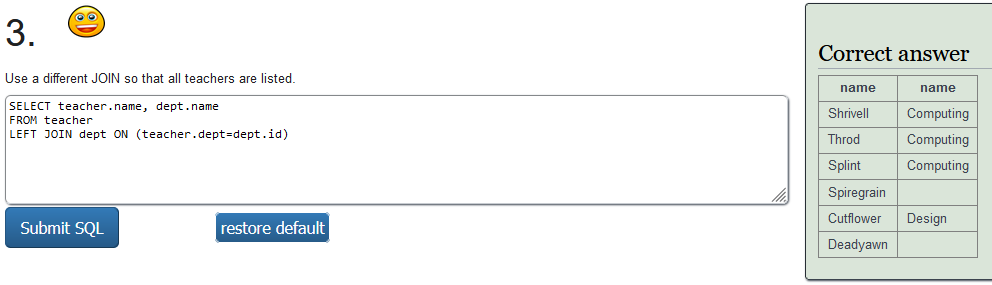
* **More JOIN Quiz:**

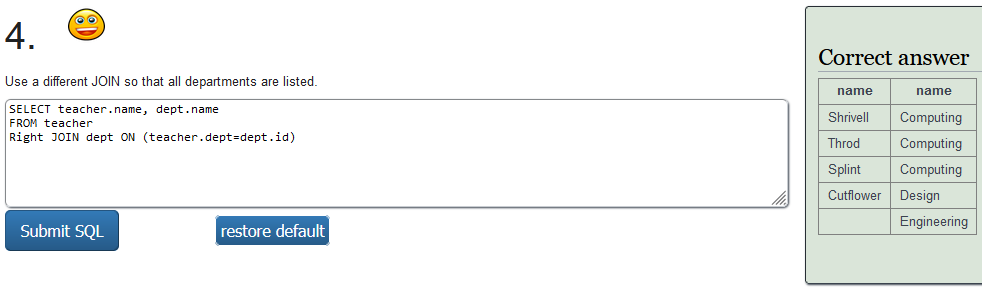


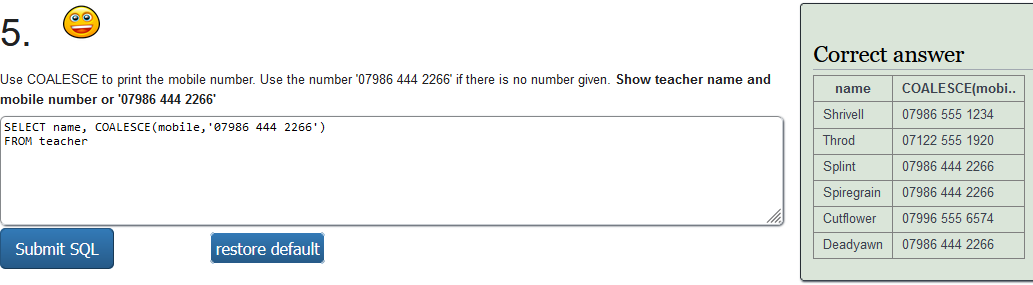
* **Using NULL:**











A screenshot of a computer

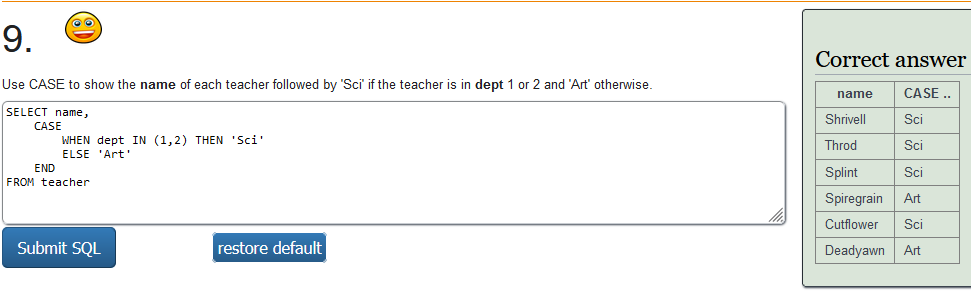
Description automatically generated

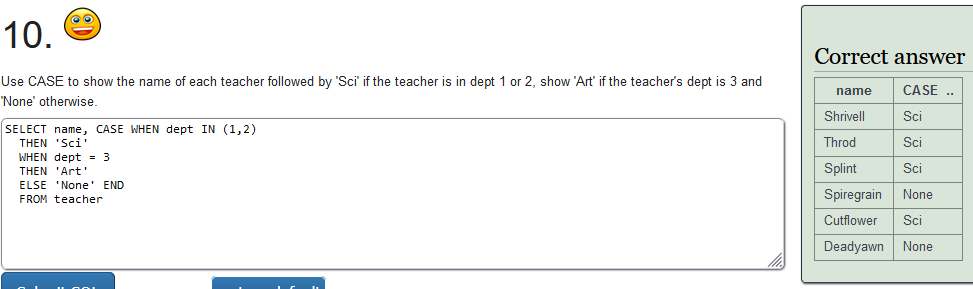
A close-up of a computer screen

Description automatically generated

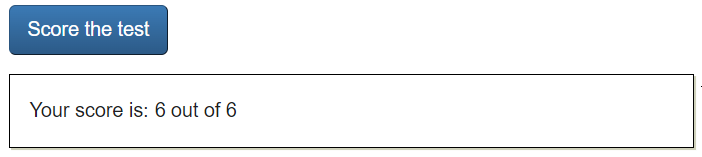
A screen shot of a computer

Description automatically generated

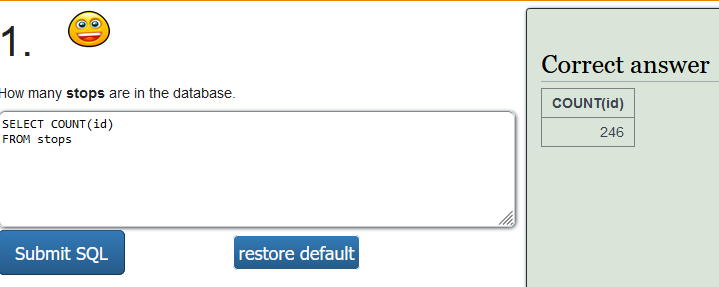




* **Using NULL Quiz:**



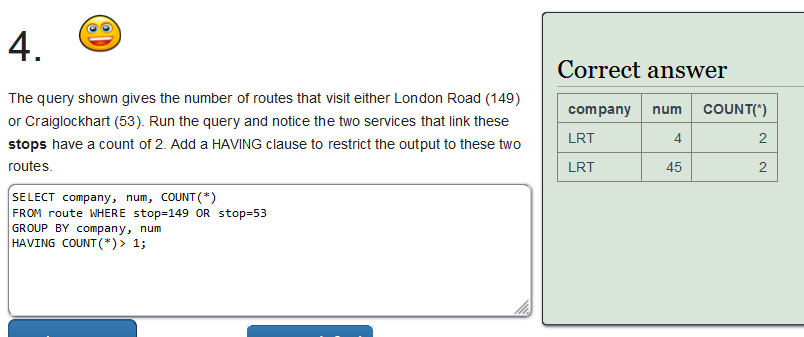
* **Self JOIN:**

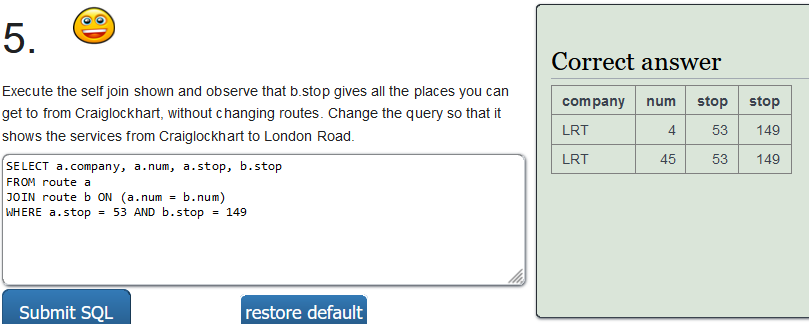
****

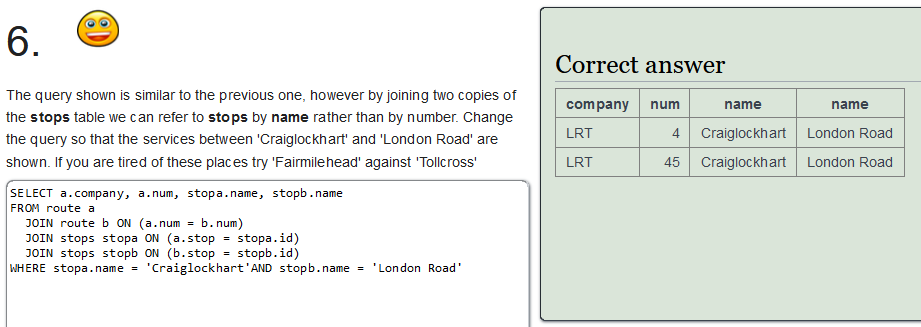
**A screenshot of a computer

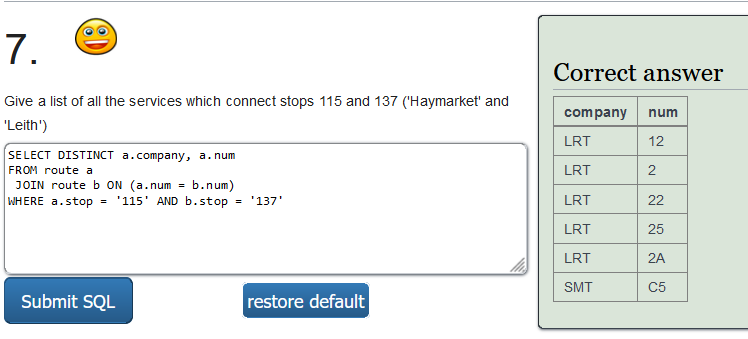
Description automatically generated**

****

****

****

****

****

**A screenshot of a computer error

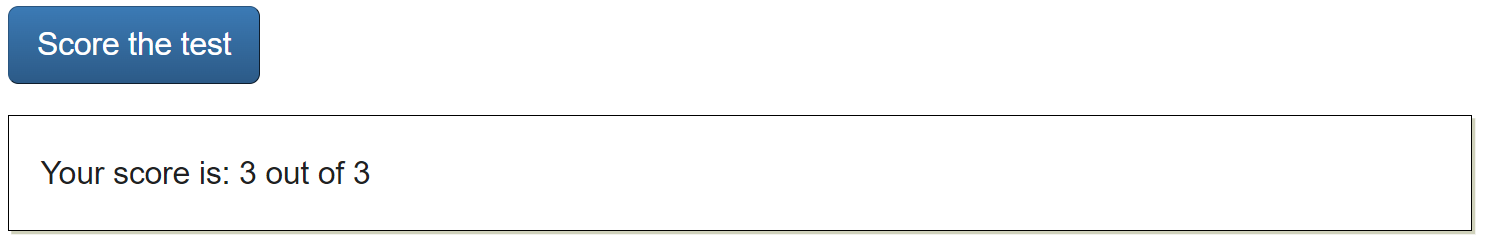
Description automatically generated**

**A screenshot of a computer

Description automatically generated**

.

* **Self JOIN quiz:**



.

1. Propongan preguntas que cumplan los siguientes requerimientos.

* **Consultas Operadores de Conjuntos (5):**

UNION:

Pregunta: ¿Cuáles son los nombres de los huéspedes que han realizado una reserva o han dejado una calificación

Consulta:

SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM Booking) UNION SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM rate);

UNION ALL:

Pregunta: ¿Cuántas veces ha sido reservada cada habitación, incluyendo las reservas repetidas?

Consulta:

SELECT room\_no, COUNT(\*) FROM Booking GROUP BY room\_no UNION ALL SELECT room\_no, COUNT(\*) FROM Booking GROUP BY room\_no;

INTERSECT:

Pregunta: ¿Cuáles son los nombres de los huéspedes que han realizado una reserva y han dejado una calificación?

Consulta:

SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM Booking) INTERSECT SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM rate);

EXCEPT:

Pregunta: ¿Cuáles son los nombres de los huéspedes que han realizado una reserva pero no han dejado una calificación?

Consulta:

SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM Booking) EXCEPT SELECT name FROM Guest WHERE GuestId IN (SELECT GuestId FROM rate);

CROSS JOIN:

Pregunta: ¿Cuál es el resultado de combinar cada habitación con cada tipo de habitación?

Consulta:

SELECT room\_no, type

FROM Booking

CROSS JOIN room\_type;

* **Consultas Operador de Junta (6):**

¿Cuáles son los nombres de los huéspedes que han realizado reservas?

SELECT DISTINCT first\_name

FROM guest INNER JOIN booking ON id = guest\_id

¿Cuáles son los nombres de los huéspedes y los números de habitación en los que han realizado reservas, incluyendo aquellos huéspedes que no han hecho ninguna reserva y las habitaciones que no tienen reservas?

SELECT Guest.name AS GuestName, Booking.room\_no AS RoomNumber

FROM Guest

LEFT OUTER JOIN Booking ON Guest.GuestId = Booking.GuestId;

¿Cuáles son los nombres de los huéspedes y los números de habitación en los que han realizado reservas, incluyendo aquellos huéspedes que no han hecho ninguna reserva pero excluyendo las habitaciones que no tienen reservas?

SELECT Guest.name AS GuestName, Booking.room\_no AS RoomNumber

FROM Guest

LEFT JOIN Booking ON Guest.GuestId = Booking.GuestId

WHERE Booking.GuestId IS NOT NULL;

¿Cuáles son los nombres de los huéspedes y los números de habitación en los que han realizado reservas, incluyendo aquellas habitaciones que no tienen reservas pero excluyendo los huéspedes que no han hecho ninguna reserva?

SELECT Guest.name AS GuestName, Booking.room\_no AS RoomNumber

FROM Guest

RIGHT JOIN Booking ON Guest.GuestId = Booking.GuestId

WHERE Guest.GuestId IS NOT NULL;

* **Consultas Operador Desconocido (2):**

¿Se quiere saber quiénes dejaron en blanco su dirección o alguna parte de su nombre?

SELECT \*

FROM guest

WHERE address OR first\_name OR last\_name IS NULL

¿Se quiere saber quiénes No gastaron ningún extra durante su estadía?

SELECT booking\_id, COALESCE(amount, 'No gasto') AS "Gasto"

FROM extra

* **Consultas Operadores Lógicos (3):**

¿Se quiere saber el nombre completo de los huéspedes alojados, cuyo nombre completo empieza por la misma letra que su dirección?

SELECT \*

FROM guest

WHERE LEFT(first\_name, 1) = LEFT(address, 1)

AND LEFT(last\_name, 1) = LEFT(address, 1)

¿Se quiere saber el nombre completo de los huéspedes alojados, donde alguna parte de su nombre completo termine por la misma letra que su dirección?

SELECT \*

FROM guest

WHERE RIGHT(first\_name, 1) = LEFT(address, 1)

OR RIGHT(last\_name, 1) = LEFT(address, 1)

¿Se quiere saber el nombre completo de los huéspedes alojados, cuyo nombre no empieza por vocales?

SELECT \*

FROM guest

WHERE NOT LEFT(first\_name, 1) AND NOT LEFT(last\_name, 1) IN ('A','E','I','O','U')

* **Consulta Operador CASE (1):**

¿Se quiere saber que huéspedes agendaron una habitación con más capacidad máxima de lo necesario?

SELECT guest\_id, room\_type,

CASE

WHEN occupants < max\_occupancy THEN 'Desaprovechada'

ELSE 'Aprovechada'

END AS Rendimiento

FROM booking JOIN room ON (booking.room\_no=room.id)

## Bibliografía

IBM. (2022). *Tipo de datos nulo (Null)* .Consultado de: [https://www.ibm.com/ docs/es/integration-bus/10.0?topic=flows-esql-null-data-type](https://www.ibm.com/%20docs/es/integration-bus/10.0?topic=flows-esql-null-data-type)

LearnSQL. (2023). *Explicación de SQL JOINs.* Consultado de: <https://learnsql.es/blog/explicacion-de-sql-joins-5-ejemplos-claros-de-inner-join-sql-para-principiantes/>

PYM. (2020). *¿Cómo funciona INNER JOIN, LEFT JOIN, RIGHT JOIN y FULL JOIN?.* Consultado de: <https://programacionymas.com/blog/como-funciona-inner-left-right-full-join>

SQLZoo. (2032). *SQL Tutorial.* Consultado de:<https://sqlzoo.net/wiki/SQL_Tutorial>

Wikipedia. (2022). *Null (SQL).* Consultado de: <https://es.wikipedia.org/wiki/Null_(SQL)>

Wikipedia. (2023). *Sentencia JOIN en SQL.* Consultado de: <https://es.wikipedia.org/wiki/Sentencia_JOIN_en_SQL>